Tatyana Vlaskin
Final Project
I made a web based mobile application that can be accessed:
http://vlaskin-assignment2.appspot.com/
Say yes to geolocation, it might work for you, but it has an attitude, does not work all the time, code for that feature was taken from 2 of the references. Overall, my mobile portion is based on the youtube tutorial. I will never be a mobile developer, to write one line of code it was taking me over an hour to figure out what was going on.

I've also made a video to demonstrate that the design was polished for mobile interface. Video can be found on
http://web.engr.oregonstate.edu/~vlaskint/bandicam%202016-03-13%2023-11-06-244.avi

Please note, I have not created accounts in the video because I already have 4 google accounts and did not feel like making more accounts.

ONE BUG- THERE HAS TO BE AT LEAST ONE EMPLOYEE ADDED BEFORE YOU CAN USE ADD CHECK FEATURE, YOU WILL GET AN ERROR IF YOU TRY TO USE IT BEFORE ADDING AN EMPLOYEE.

An explanation of the URI structure used to access resources

This project is similar to assignment 3. But instead of dishes and reviews, we have employees and checks. If you look at the code, I have not changed the variables. **Dishes are employees and reviews are checks.** One thing that was added is user, in this report, I did changed variables, but not in the code.

There are two entity classes

```
class Employee(ndb.Model):
    name = ndb.StringProperty()
    phone_number = ndb.StringProperty()
    email = ndb.StringProperty()
//as you can see when employee is added, its linked to the user
```

```
  user = ndb.UserProperty()
```

So as we can see when the employee is added, its linked to user, where the user comes from the user = users.get_current_user() - will be discussed later.

```
class Check(ndb.Model):
  date = ndb.StringProperty()
  check_amount = ndb.FloatProperty()
  employee_link = ndb.KeyProperty(kind = 'Employee')
```

Please note that I am using Method = post with different values, except when the data is displayed

GET to view employees pulls all the employees from the database
dish.py and /view2
GET to view checks- works only when there are employees in the database- pulls all the checks from the database associated with employees
/17 and dish.py
POST to add employees add employee to the database
/add2 and dish.py
POST to add checks -- please note there is a bug here, this will work only if there are already employees in the database. Add check to the database
/add and dish.py
PUT employees get information about the employees and updates it
/edit2 /view2 and dish.py
PUT checks get information about the checks and updates it
/edit /viewand dish.py
DELETE employees deletes employee and all checks connected to that employee
/view2 and dish.py
DELETE checks - works only where there are employees in the database. Deletes check for the employee
/17 and dish.py

USER ACCOUNT:
For the user account I am using the Google Accounts authentication. Using the Google Accounts authentication, my app is detecting whether the current user has signed in. If

the user is signed in to the app, the app can access the user email address, as well as a unique user ID and use all the features of the app.

If the user is not signed in and the app is started, the user will be redirected to the appropriate sign in page. In case they do not have a google account, they would need to make one and after that they will be able to sign in to the app. This is handled with the following code:

```
class MainPage(webapp2.RequestHandler):
    def get(self):
        user = users.get_current_user()
        if user:
            user_email = user.email()
            #logout = util.create_logout_url('')
            #logout = users.create_logout_url('/')
            template_values = {
                'logout' : util.create_logout_url(''),
            }
            template = JINJA_ENVIRONMENT.get_template('index.html')
            self.response.write(template.render(template_values))
        else:
            self.redirect(users.create_login_url(self.request.uri))
```

All the pages require the user to be signed in order to access and this is enforced in the app.yaml file.

- url: /.*
script: dish.application
login: required

So what happens If a user accesses any of the above mentioned URL, the app,yaml is configured in such a way that it requires sign-in and after the user is signed in they are redirected to the index.html page, which will list all the options of the app.

So the question: is it safe?

We place all the trust in Google by using their Authenticator. Everybody is using it, so it must be safe. Here is the list of the issues that can be identified with the google Authenticator:

1. The current version of the google authenticator is proprietary. In security, when you hide what you're doing, the near-universal rule is that you're hiding your bugs and backdoors from the world. Nobody can, or should, trust you right there. It's just common sense.
2. They never ask for password update, which is bad practice when it comes to security
3. At the same time Google Authenticator is a client app that have updates. Thus, there might be an update, which actually has happened already, such that all the users who updated their apps and it did not remember their passwords anymore. That meant that users were lockced out of their accounts Refferebnce: http://www.eweek.com/blogs/security-watch/password-lies-truth-and-google-authenticator.html

Constraints met for a Restful API:

Stateless:

The communication between service consumer (client) and service (server) must be stateless between requests. This means that each request from a service consumer should contain all the necessary information for the service to understand the meaning of the request, and all session state data should then be returned to the service consumer at the end of each request.

I was able to meet this requirement as well. All the requests done by user have all the necessary information to fulfill the request [either add, or delete, or edit, or view] and in the body of each request, all parameters and data that are needed to generate a response are sent to the server.

Client-server: The Client-Server constraint requires that a service offer one or more capabilities and listen for requests on these capabilities. A consumer invokes a capability by sending the corresponding request message, and the service either rejects the request or performs the requested task before sending a response message back to the consumer.

I think I met requirement. User is able to add, remove, edit both employees and checks, so user is able to send a corresponding message and gets an answer back from the server. As for the server, it is able to listen to the requests from the clients and can send responds, by adding, editing, deleting or displaying data back to the client.
Overall client does not care how the data is stored and server does not care about the interface. The server is able to interface with multiple clients. As for the clients, they do see and they do not needs to see what's going on the server. These was achieved using google-app engine.

Layered System

A REST-based solution can be comprised of multiple architectural layers, and no one layer can "see past" the next. Layers can be added, removed, modified, or reordered in response to how the solution needs to evolve. **This deals with the separation of concerns, every layer in my app has a single purpose and deal only with that. For example when the user wants to see the information about the employer, the server only returns values [name, email, telephone number] and the client decides how to presents this data, the html form are the decisions point how the data is presented to the client. I think I met this requirement**

Uniform Interface:

Separation of resources from representations. Resources are the capabilities of the server, while their representation are what the resources provides to the client. Uniform interface should identify message behavior without looking in the message body. The URL should state the resource being manipulated. Uniform interfaces do not depend on saved message states. Uniform interface rely on standard messages (GET, HEAD, POST, DELETE) and communicate media processing information. **In my app, a resource is the py script that runs on the server, and the representations are the HTML files with all the requested information that the server returns when a browsers make a request. The browsers never gets my py files, all it gets is the HTML files. Python code is not visible in my app. My interface relies on the message standard [i am using POST}, but at the same time it depends on the saves message state, meaning, I have POST messages with different values [delete, add, etc] and if I understand it correctly this should not be present in the uniform interface.**

Cache

Response messages from the service to its consumers are explicitly labeled as cacheable or non-cacheable. This way, the service, the consumer, or one of the intermediary middleware components can cache the response for reuse in later requests.

I have not meet this requirement at all.

MOBILE FEATURE:

Social media login

I think that google is the social media and I am using it to sign in into my app.

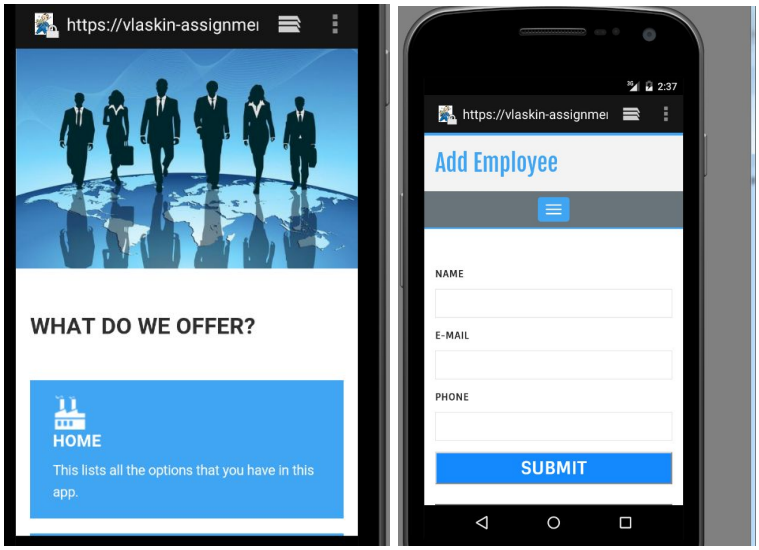
One Click Email

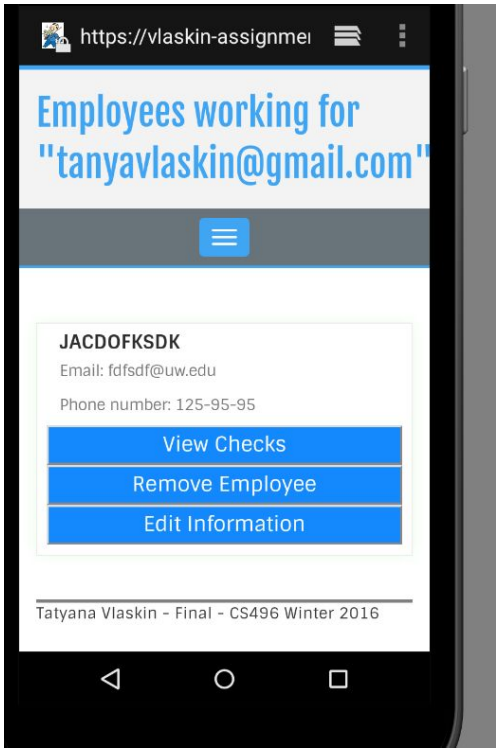
By clicking a button, you are taken directly you google email, which can be used to send message

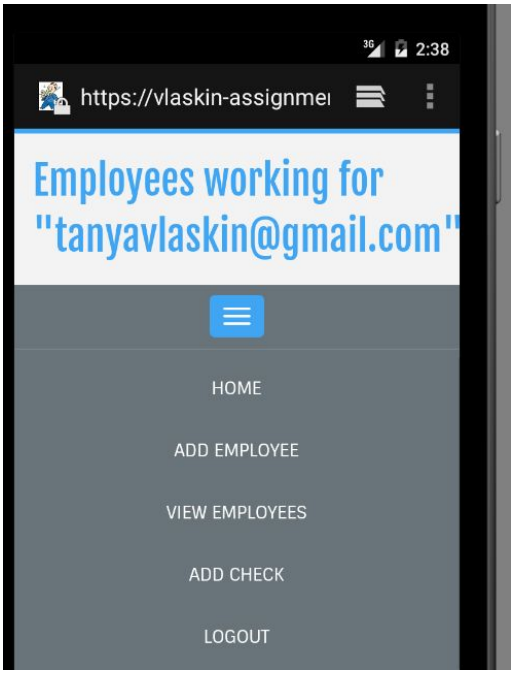
Responsive Design

My app adjust layout dynamically according to the screen dimensions of the device it's running on.

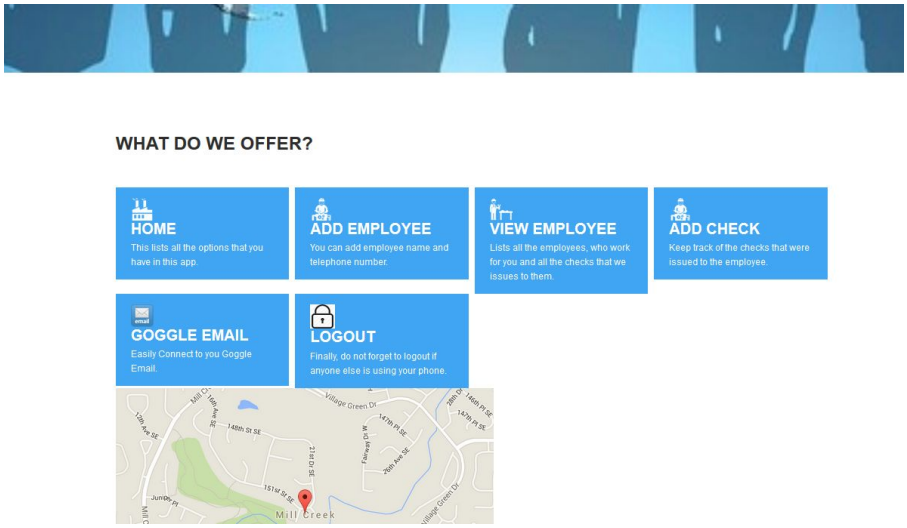
Mobile view







Website View:



Geolocation

I've tried to add this feature, but the problem is that it does not work consistently, it seems it has an attitude.

References [please also see references for assignment 3b]

http://stackoverflow.com/questions/18100442/samsung-galaxy-devices-cant-use-geolocation-getcurrentposition

https://www.pubnub.com/blog/2015-05-13-displaying-android-location-live-updating-map-google-maps/

http://tympanus.net/codrops/2012/03/27/login-and-registration-form-with-html5-and-css3/

https://www.youtube.com/watch?v=NU_wNR_UUn4

https://developers.google.com/identity/protocols/OAuth2

https://w3layouts.com/preview/?l=/steel-industry-a-industrial-category-flat-bootstrap-responsive-web-template/

http://www.markwedgworth.com/Employee-Benefits-Expertise.9.htm

https://www.youtube.com/watch?v=5v_1cqkFSuQ

https://www.youtube.com/watch?v=Rqy_G3tY62c

https://www.youtube.com/watch?v=EyQ7PcQ8ViU

https://www.youtube.com/watch?v=MwYaYlsWiPE

https://www.youtube.com/watch?v=JynS3xNKtto

http://www.slideshare.net/zomigi/building-responsive-layouts-15508821

http://getbootstrap.com/2.3.2/

http://jquery.com/download/

https://ctrlq.org/code/19724-find-people-location

http://stackoverflow.com/questions/7810607/google-app-engine-oauth2-provider

http://stackoverflow.com/questions/18310434/how-to-set-text-size-in-a-button-in-html

https://w3layouts.com/preview/?l=/steel-industry-a-industrial-category-flat-bootstrap-responsive-web-template/

http://stackoverflow.com/questions/4324843/how-should-i-write-my-google-app-engine-app-yaml-file

https://github.com/googleads/googleads-dotnet-lib/wiki/How-to-create-OAuth2-client-id-and-secret

https://support.google.com/cloud/answer/6158849?hl=en&ref_topic=6262490

https://support.google.com/cloud/answer/6158857?hl=en

https://github.com/2cloud/AppEngine/tree/4cd927f66e3b552a8997e622df97d0a359c80615

https://stackoverflow.com/questions/13657506/google-app-engine-redirect-to-user-login-on-button-click

http://www.mysamplecode.com/2012/10/google-app-engine-python-user-authentication.html

https://cloud.google.com/appengine/docs/python/refdocs/google.appengine.api.users

https://github.com/isaachawley/shallow-coffee/blob/2ec50892090acab64a73951498f545be0507e6fc/main.py

https://books.google.com/books?id=4BIDCgAAQBAJ&pg=PA29&lpg=PA29&dq=users.create_logout_url(%27/%27)+hot+to+sent+to+template&source=bl&ots=lmGooAQexc&sig=J9AnLvyST2QMoMnAw9IbU8zHtc0&hl=en&sa=X&ved=0ahUKEwj2-ISA4LHLAhVRy2MKHdHF

ATQQ6AEINjAE#v=onepage&q=users.create_logout_url('%2F')%20hot%20to%20sent%20to%20template&f=false

https://cloud.google.com/appengine/docs/python/gettingstartedpython27/templates

https://groups.google.com/forum/#!topic/google-appengine/EcgKT6sJJZE

http://www.paulund.co.uk/how-to-use-geolocation-api-with-google-maps

http://stackoverflow.com/questions/5329662/android-webview-geolocation

http://stackoverflow.com/questions/4818157/geolocation-in-webview-for-android

http://stackoverflow.com/questions/11280946/unable-to-get-gps-coordinates-using-javascript-on-browser-in-android/11298468#11298468

https://turbomanage.wordpress.com/2012/04/23/how-to-enable-geolocation-in-a-webview-android/

http://www.codota.com/android/scenarios/52fcbd77da0a1ca94ad37a0d/android.webkit.WebView?tag=dragonfly

https://gist.github.com/Cheesebaron/ad84740c9bffa7e255c8

http://tjach.pl/tresc/uploads/2014/11/sm_wyk6.pdf